

# ICONS 2007 - Tutorial Proposal

## Bridging Requirements, Use Cases and Object-Oriented Modeling for Systems Engineering

### Contact Details:

<b>Name(s):</b>	Prof. Dr. Hermann Kaindl
<b>Employer:</b>	Vienna University of Technology Institute of Computer Technology
<b>Address:</b>	Gusshausstr. 27-29 A-1040 Vienna, Austria
<b>Telephone:</b>	+43 1 58801-38416
<b>Fax:</b>	+43 1 58801-38499
<b>E-mail:</b>	<a href="mailto:kaindl@ict.tuwien.ac.at">kaindl@ict.tuwien.ac.at</a>

### Instructor Details

<b>Presenters Biography:</b>	<p><b>Prof. Hermann Kaindl</b> joined the Institute of Computer Technology at the Vienna University of Technology in Vienna, Austria, in early 2003. Prior to moving to academia as a full professor, he was a senior consultant with the division of program and systems engineering at Siemens AG Austria. There he has gained more than 24 years of industrial experience in software development. His current research interests include software and systems engineering with a focus on requirements engineering, and human-computer interaction as it relates to scenario-based design. He has published four books and more than ninety papers in refereed journals, books and conference proceedings. He is a senior member of the IEEE, a member of the ACM and the INCOSE, and is on the executive board of the Austrian Society for Artificial Intelligence.</p>
<b>Past Presentation Experience:</b>	<p>This tutorial has matured over several years:</p> <ul style="list-style-type: none"><li>• “<a href="#">Reconciling Requirements, Use Cases and Object-Oriented Modeling</a>” at RE’02 (17 attendees in a conference of about 200)</li><li>• “<a href="#">Reconciling Business Modeling and Requirements with Object-Oriented Software Development</a>” at HICSS’36 (38 attendees in a conference of about 600)</li><li>• “<a href="#">Reconciling Requirements, Use Cases and Object-Oriented Modeling for Systems Engineering</a>” at INCOSE 2003 (37 attendees in a conference of about 1000).</li></ul>

	<ul style="list-style-type: none"> <li>• <a href="#">“Reconciling Requirements, Use Cases and Object-Oriented Modeling for Systems Engineering”</a> at INCOSE 2004 (26 attendees in a conference of about 850)</li> <li>• <a href="#">“Modeling Business and Requirements Using UML”</a> at HICSS’38 (35 attendees in a conference of about 600)</li> <li>• <a href="#">“Seamless Use of Object-Oriented Models from Requirements to Software Design”</a> at OOPSLA 2006 (15 attendees in a conference of about 1100)</li> <li>• <a href="#">“From Business Models and Requirements to Architectural Software Design”</a> to be presented at HICSS’40</li> </ul> <p>In addition, this proposer has previously held tutorials on the reuse of requirements at CAiSE'00, RE'01, RE'03, INCOSE 2004, RE'04 and INCOSE 2005.</p>
--	--

## Tutorial Details

<b>Tutorial Title:</b>	<b>Bridging Requirements, Use Cases and Object-Oriented Modeling for Systems Engineering</b>
<b>Duration:</b>	Half Day
<b>Class Size – Max:</b>	None
<b>Class Size – Min:</b>	Whatever is financially viable.
<b>Technical Abstract:</b>	<ul style="list-style-type: none"> <li>• How do scenarios / use cases fit together with functional requirements?</li> <li>• How can OO (object-oriented) principles like classification help organizing a huge number of requirements?</li> <li>• How can the application domain be better understood using OO modeling?</li> </ul> <p>This tutorial addresses these questions because they are relevant for industrial software development but too many misunderstandings still exist with regard to OO processes and methods as related to requirements engineering. It shows how each requirement given in natural language can be modeled as an object, which facilitates a clean organization and association. While scenarios / use cases can somehow illustrate the overall functionality, additionally functional requirements for the system to be built should be formulated and related to them appropriately. In order to better understand scenarios, the goals to be achieved by them should be explicitly defined and linked to them as well. All kinds of requirements typically make statements about the application domain, which should be represented in an OO Domain Model of conceptual classes, in order to make the requirements better understandable.</p>

	<p>Among other things, this tutorial proposes solutions to issues discussed in a panel organized by this proposer at OOPSLA 2001 “<a href="#">How do Requirements Relate to Objects?</a>” and another panel with the same title at <a href="#">INCOSE 2004</a>. It includes also material on real-world experience from the approach developed by this proposer as presented in an invited State-of-the-Practice Talk at RE'01:</p> <p>H. Kaindl, <a href="#">Adoption of Requirements Engineering: Conditions for Success</a>, <i>Fifth IEEE International Symposium on Requirements Engineering (RE'01)</i>, Toronto, Canada, August 2001.</p> <p>It is based on an in-house course at Siemens, a teaching course at the Vienna University of Technology, research and consulting experience of its proposer and, e.g., on the following articles and papers:</p> <p>H. Kaindl, How to Identify Binary Relations for Domain Models, In <i>Proceedings of the Eighteenth International Conference on Software Engineering (ICSE-18)</i>, IEEE Computer Society Press, Los Alamitos, CA, 1996, pp. 28–36.</p> <p>H. Kaindl, A Practical Approach to Combining Requirements Definition and Object-Oriented Analysis, <i>Annals of Software Engineering</i>, Vol. 3, 1997, pp. 319-343.</p> <p>H. Kaindl, Difficulties in the transition from OO analysis to design, <i>IEEE Software</i>, Sept./Oct. 1999, pp. 94-102.</p> <p>H. Kaindl, A Design Process Based on a Model Combining Scenarios with Goals and Functions, <i>IEEE Transactions on Systems, Man, and Cybernetics (SMC)</i>, Part A 30(5), Oct. 2001, pp. 537-551.</p> <p>H. Kaindl, Is object-oriented requirements engineering of interest?, <i>Requirements Engineering</i>, Vol. 10, 2005, pp. 81–84.</p> <p>H. Kaindl, A Scenario-Based Approach for Requirements Engineering: Experience in a Telecommunication Software Development Project, <i>Systems Engineering</i>, Vol. 8, 2005, pp. 197–210.</p>
<p><b>Detailed Tutorial Outline:</b></p>	<ul style="list-style-type: none"> <li>• <i>Background</i> <ul style="list-style-type: none"> <li>- Requirements</li> <li>- Use cases</li> <li>- Object-oriented modeling features</li> <li>- Iterative and incremental development</li> </ul> </li> <li>• <i>Functional requirements, goals and scenarios / use cases</i> <ul style="list-style-type: none"> <li>- Relation between scenarios and functions</li> <li>- Relation between goals and scenarios</li> <li>- Composition of these relations</li> </ul> </li> <li>• <i>Requirements and object-oriented models</i></li> </ul>

	<ul style="list-style-type: none"> <li>- Metamodel in UML</li> <li>- Requirements and objects</li> <li>- Taxonomy of requirements</li> </ul> <ul style="list-style-type: none"> <li>• <i>Systematic process</i> <ul style="list-style-type: none"> <li>- Navigation in the metamodel graph</li> <li>- Partial sequences of steps</li> <li>- Improvements through this process</li> </ul> </li> <li>• <i>Final discussion</i></li> </ul>
<b>Intended Audience:</b>	Systems engineers, requirements engineers, software engineers, project managers.
<b>Tutorial Objectives:</b>	<p>The participants will understand several important issues with regard to object-oriented approaches that are relevant for enterprise modeling and systems development. They will experience UML as a language for representing OO models, but also the need to be clear about what kind of objects are represented. In addition, participants will see how scenarios and use cases can be utilized for requirements engineering. But they will also see the additional need to specify the functional requirements for the system to be built.</p> <p>The purpose of this proposed tutorial is to bridge requirements engineering and object-oriented modeling, so that practitioners can apply the best from both “worlds” together for systems engineering.</p>
<b>Audience Prerequisites:</b>	The assumed attendee background is some familiarity with scenarios / use cases and basic object-oriented concepts, as well as interest in requirements.
<b>Tutorial Presentation Details</b>	<p>PowerPoint Presentation, group exercises</p> <p>For the lectures, the instructor will use a computer screen projector connected with a laptop computer.</p>
<b>Tutorial Handout Details:</b>	Tutorial Presentation Notes plus a selection of my published papers that this tutorial is based upon.